# Comparative Study of Word Alignment Heuristics and Phrase-Based SMT

**Hua Wu** and **Haifeng Wang**

Toshiba (China) Research and Development Center
5/F., Tower W2, Oriental Plaza
No.1, East Chang An Ave., Dong Cheng District
Beijing, 100738, China
{wuhua, wanghaifeng}@rdc.toshiba.com.cn

**Abstract**

This paper comparatively analyzes six different word alignment heuristics and their impacts on translation quality. We also propose a method to filter the noise in the phrase tables extracted by these heuristic methods and examine the effectiveness of combination of the methods. Experiments are performed on the Europarl corpus, where a multilingual in-domain training corpus, an in-domain test set, and an out-of-domain test set are available. Results indicate that (1) the heuristics show similar tendencies in the word alignment task on both test sets, but they perform differently in the translation task on the in-domain and out-of-domain test sets; (2) in general, the relationship between word alignment and machine translation performance is difficult to be predicted, depending on domains of the training and testing corpora besides other factors such as evaluation metrics and the characteristics of translation systems; (3) noise filtering and combination of these heuristic methods achieve larger improvement on the out-of-domain test set than on the in-domain test set.

## Introduction

Word or phrase alignment plays a crucial role in statistical machine translation (SMT). During training, the SMT systems produce alignment between words or phrases of existing examples to estimate the statistical parameters. With these estimated parameters, the SMT systems translate source sentences into target sentences.

Current state-of-the-art models in machine translation are based on alignments between phrases (Koehn et al., 2003; Chiang, 2005). Phrase-based generative models are first proposed by Marcu and Wong (2002) to extract phrase pairs. Zhao and Waibel (2005) also proposed several generative models to generate phrase pairs for machine translation. An alternative is to first generate word alignments. Phrase alignments are then inferred heuristically from these word alignments (Och et al., 1999; Koehn et al., 2003). DeNero et al. (2006) showed in their experiments that the heuristic methods outperform the generative models. Their analysis indicates that the performance gap stems primarily from the segmentation variable of the generative model, which increases the possibility of overfitting during training.

Recently, several researches have been conducted to explore the relationship of word alignment quality measures and machine translation quality. The main points are concluded as follows.

1. It is difficult to find a direct correlation between word alignment measures (such as alignment error rate) and automated MT metrics (Ayan and Dorr, 2006; Fraser and Marcu, 2006).
2. Large gains in alignment performance under any metric are confirmed to achieve relatively small gains in translation performance (Lopez and Resnik, 2006).
3. Better feature mining can lead to substantial gain in translation quality (Lopez and Resnik, 2006).
4. It is better to generate alignments adapted to the characteristics of the translation models that will make use of this alignment information (Vilar et al., 2006).

However, all of the above conclusions are made on the in-domain test sets and never on the out-of-domain test sets. In addition, although Lopez and Resnik (2006) pointed out that it may be more useful to handle noise in phrase extraction than to improve word alignment quality, they did not provide detailed information to verify this point.

In this paper, we will use different heuristics to generate word alignments, and examine the impacts of these heuristics on machine translation quality. And then we will re-evaluate the relationship of word alignment and their impacts on machine translation quality on both in-domain and out-of-domain test sets.

Furthermore, in order to examine the noise in the phrase pairs extracted using different alignment heuristics, we propose a method to filter the noise in the phrase tables using association measures. And we will also investigate whether combining the phrase tables extracted by different heuristics improves translation quality.

We performed experiments on the Europarl corpus (Koehn, 2005; Koehn and Monz, 2006), where a multilingual in-domain training corpus, an in-domain test set, and an out-of-domain test set are available. We obtained the following results:

1. Word alignment results show that the *compromise method*, which makes compromise between precision and recall, performs the best on both in-domain and out-of-domain alignment test sets.
2. Translation results indicate that the heuristic methods perform differently on the in-domain and out-of-domain test sets. On the in-domain test set, the recall-oriented heuristic methods yield better translation quality. On the out-of-domain test set, the precision-oriented heuristic methods yield better translation quality. On both of the test sets, the compromise method achieves satisfying translation quality.
3. The relationship between word alignment and machine translation performance depends on domains of the training and testing corpora besides other factors such as evaluation metrics and the characteristics of the translation systems used.
4. Filtering the noise in the phrase tables and combining different phrase tables achieve larger improvement on the out-of-domain test set than on the in-domain test set.

The remainder of this paper is organized as follows. First, we will describe phrase-based machine translation and the corresponding word alignment heuristics used in this paper. Then we will propose a method to filter the noise in phrase pairs. Following this, we will propose methods to combine the phrase pairs extracted by different methods. After that, we will present the experimental results. Lastly, we will conclude this paper.

## Phrase-Based Statistical Machine Translation

In phrase-based SMT systems, the unit of translation is any contiguous sequence words, which is called phrase. It includes two steps: training and translation. During training, parallel corpus is employed to induce phrase alignment in the sentence pairs and estimate translation probabilities. Target monolingual corpus is employed to train a language model. During translation, the source sentence is first segmented into phrases and then translated into target phrases using learned phrase pairs. The target phrases are then recombined to form a target sentence.

### Log-Linear Model

Given a source sentence $\mathbf{f}$, the best target translation $\mathbf{e}_{\text{best}}$ can be obtained according to the following log-linear model

$$\begin{aligned}\mathbf{e}_{\text{best}} &= \arg\max_{\mathbf{e}} p(\mathbf{e}\mid\mathbf{f}) \\ &\approx \arg\max_{\mathbf{e}} \sum_{m=1}^{M} \lambda_m h_m(\mathbf{e},\mathbf{f})\end{aligned} \quad (1)$$

Where $h_m(\mathbf{e},\mathbf{f})$ represents feature functions, and $\lambda_m$ is the weight assigned to the corresponding feature function. In this paper, we will use the Pharaoh system (Koehn, 2004). Eight different features are used in this system.

1. a phrase translation probability
2. an inverse phrase translation probability
3. a lexical weight: measuring the quality of word alignment inside the phrase pair
4. an inverse lexical weight
5. language model
6. phrase penalty
7. word penalty
8. reordering

For phrase translation probability, lexical weight, and reordering, we use the same models in (Koehn et al., 2003). We use n-grams for language modelling. For the phrase penalty and word penalty, we use the same heuristics in (Zen and Ney, 2004).

### Word Alignment Heuristics

One important component used in the Pharaoh system is the phrase translation table. Since DeNero et al. (2006) showed in their experiments that the heuristic methods outperform the generative models for phrase pair extraction, we use heuristic methods in this paper. We first align the words in the training parallel corpus, extract phrase pairs that are consistent with the word alignments, and then assign probabilities to the obtained phrase pairs.

Word alignments are obtained by using the GIZA++ toolkit[1] in both translation directions and then symmetrize the two alignments. In statistical translation models implemented in GIZA++, only one-to-one and more-to-one word alignment links can be found. Thus, some multi-word units cannot be correctly aligned. The symmetrization method is used to effectively overcome this deficiency (Och and Ney, 2003).

In this paper, we use six kinds of symmetrization methods. Let $A_1$ and $A_2$ represent the two alignments in source to target and target to source translation directions, the six symmetrization methods can be described as follows.

1. intersection: $A = A_1 \cap A_2$
2. union: $A = A_1 \cup A_2$
3. grow: the alignments in the intersection set of the two alignments are first added. And then neighboring alignment points in the union sets directly in the left, right, top, or bottom directions are added.
4. grow-diag: besides the neighboring points in the grow method, the diagonally neighboring alignment points are also included.
5. grow-diag-final: in addition to the alignment points in grow-diag, the non-neighboring alignment points between words, of which at least one is currently unaligned, are added in a final step.
6. grow-final: In addition to the alignment points in grow, the non-neighboring alignment points between words, of which at least one is currently unaligned, are added in a final step.

### Phrase Extraction

With the word alignment results obtained by the above six heuristic methods, we extract phrase pairs that satisfy the following restrictions:

1. all source words within a phrase are aligned only to target words within a phrase
2. all target words within a phrase are aligned only to source words within a phrase

More formally, the set of bilingual phrases consistent with a word alignment $A$ is defined as

$$\begin{aligned}&BP(f_1^J, e_1^I, A) \\ &= \{(f_j^{j+m}, e_i^{i+n}) \mid \forall (i', j') \in A: \\ &\qquad j \le j' \le j+m \Leftrightarrow i \le i' \le i+n\}\end{aligned} \quad (2)$$

The phrase translation probability is defined as

$$p(\bar{f} \mid \bar{e}) = \frac{count(\bar{f}, \bar{e})}{\sum_{\bar{f}'} count(\bar{f}', \bar{e})} \quad (3)$$

Where $count(\bar{f}, \bar{e})$ describes the frequency of the phrase $\bar{f}$ is aligned with the phrase $\bar{e}$ in the parallel corpus.

Given a phrase pair $(\bar{f}, \bar{e})$ and a word alignment $a$ between the source word positions $i = 1,...,n$ and the target word positions $j = 1,...,m$, the lexical weight can be estimated according to the following method (Koehn et al., 2003).

[1] It is located at http://www.fjoch.com/ GIZA++.html.

$$p_w(\bar{f} \mid \bar{e}, a) = \prod_{i=1}^{n} \frac{1}{|j \mid (i,j) \in a|} \sum_{\forall (i,j) \in a} w(f_i \mid e_j) \quad (4)$$

## Noise Filtering of Phrase Pairs

Phrase translation probability and lexical weight are important features in the phrase translation table. Lopez and Resnik (2006) found that alignment quality has little impact on the lexical weighting feature, which itself provides only a modest improvement in translation quality. Thus, we only filter the phrase pairs using phrase translation statistics.

Although the phrase translation probability described in equation (2) can be used to filter the phrase table, translation probability usually overestimates the infrequently occurring pairs. In order to solve this problem, we use association measures to filter some phrase pairs. Dunning (1993) proved that log likelihood ratio performed very well on infrequently occurring data. Thus, we calculate the log likelihood ratio for each phrase pair. First we construct a contingency table as shown in Table 1.

| | Target phrase | ~Target phrase | Totals |
|---|---|---|---|
| Source phrase | $n_{11}$ | $n_{12}$ | $R_1$ |
| ~Source phrase | $n_{21}$ | $n_{22}$ | $R_2$ |
| Totals | $C_1$ | $C_2$ | N |

Table 1. Contingency Table for Phrase Pairs

According to the contingency table, the log likelihood ratio for each phrase pair is defined as

$$G^2(\bar{f}, \bar{e}) = -2 \log \lambda = \sum_{i,j} n_{ij} \log \frac{n_{ij} N}{R_i C_j} \quad (5)$$

For each source phrase, it may be translated to n $n (n \geq 1)$ target phrases. For these $n$ phrase pair, we can obtain the maximum log likelihood value as follows:

$$Max(\bar{f}) = \max_{e} G^2(\bar{f}, \bar{e}) \quad (6)$$

As compared with the maximum value, we can get a relative value as described in (7). We only keep those phrase pairs whose relative values are larger than a threshold.[2]

$$Ratio(\bar{f}, \bar{e}) = \frac{G^2(\bar{f}, \bar{e})}{Max(\bar{f})} \quad (7)$$

## Model Combination

### Model Interpolation

To combine the different phrase tables, we use linear interpolation method in this paper. For the phrase translation probability and lexical weight in the translation models, we interpolate them as shown in equations (8) and (9).

$$p(\bar{f} \mid \bar{e}) = \sum_{i=1}^{n} \alpha_i p_i(\bar{f} \mid \bar{e}) \quad (8)$$

$$p_w(\bar{f} \mid \bar{e}, a) = \sum_{i=1}^{n} \beta_i p_{w,i}(\bar{f} \mid \bar{e}, a) \quad (9)$$

Where $p_i(\bar{f} \mid \bar{e})$ and $p_{w,i}(\bar{f} \mid \bar{e}, a)$ $(i = 1,...,n)$ are the phrase translation probability and lexical weight estimated by $n$ different methods. $\alpha_i$ and $\beta_i$ are interpolation coefficients, ensuring $\sum_{i=0}^{n} \alpha_i = 1$ and $\sum_{i=0}^{n} \beta_i = 1$.

### Count Merging

Another way to combine the phrase pairs extracted by different methods is to use the count merging method, which is widely used in language modeling (Bacchiani and Roark, 2003; Bacchiani et al., 2004). The main idea of count merging is to assign weights to the occurring count of phrase pairs, and then merging them to build translation models. The method to estimate the translation probability is shown in equation (10).

$$p(\bar{f} \mid \bar{e}) = \frac{\sum_{i=1}^{n} \alpha_i count_i(\bar{f}, \bar{e})}{\sum_{i=1}^{n} \alpha_i \sum_{\bar{e}'} count_i(\bar{f}, \bar{e}')} \quad (10)$$

Where $count_i(\bar{f}, \bar{e})$ describes the frequency of the phrase $\bar{f}$ aligned with the phrase $\bar{e}$ of the $i$th method. $\alpha_i$ is the weight assigned to the corresponding method.

For the lexical weight, we first get the lexical translation probability as shown in (11), and then calculate the lexical weight as shown in equation (4). When calculating the lexical weight, the word alignment information can be set as the union of the $n$ alignments involved.

$$w(f \mid e) = \frac{\sum_{i=1}^{n} \beta_i count_i(f, e)}{\sum_{i=1}^{n} \beta_i \sum_{e'} count_i(f, e')} \quad (11)$$

Where $count_i(f, e)$ describes the frequency of the word $f$ is aligned with the word $e$ of the $i$th method. $\beta_i$ is the weight assigned to the corresponding method.

## Experiments on Word Alignment and Translation

This section first describes the word alignment and translation results, and then analyzes the relationship between word alignment methods and machine translation quality.

[2] This threshold is determined on a development set.

## Corpus Description

### Translation Data

A shared task to evaluate machine translation performance was organized as part of the NAACL/HLT 2006 Workshop on Statistical Machine Translation (Koehn and Monz, 2006). The shared task used the Europarl corpus (Koehn, 2005), in which four languages are involved: English, French, Spanish, and German. The shared task performed translation between English and the other three languages. In our work, we perform translation from the other three languages to English.

Table 2 shows the information about the bilingual training data. In the table, "fr", "en", "es", and "de" denotes "French", "English", "Spanish", and "German", respectively.

| Language pairs | Sentence pairs | Source words | Target words |
|---|---|---|---|
| fr-en | 688,031 | 15,323,737 | 13,808,104 |
| es-en | 730,740 | 15,676,710 | 15,222,105 |
| de-en | 751,088 | 15,256,793 | 16,052,269 |

Table 2. Training Corpus for European Languages

For the language models, we use the same data provided in the shared task. We also use the same development set and test set provided by the shared task. The in-domain test set includes 2,000 sentences and the out-of-domain test set includes 1,064 sentences for each language.

### Word Alignment Data

The training data for word alignment is the same as that used for translation. For the in-domain test set, we use the Spanish-English European Parliament Plenary Sessions (EPPS) test set,[3] which is extracted from the proceedings of the European Parliament. It includes 500 sentences of at most 100 words that have been selected at random from the English-Spanish training corpus. The data set has been split into a 100 sentence pairs development corpus and a 400 sentence pairs test corpus. In our experiments, we use the same 400 sentence pairs as the test set. The test set was aligned manually by agreement of three manual reference alignments (Lambert et al., 2005). It includes 17,474 reference alignment links. 66.7% of them are sure links whereas 33.3% are possible links.

For the out-of-domain test set, we randomly extract 395 sentence pairs from the out-of-domain translation test set described in the above section. This set is also manually annotated, but we do not classify it into sure or possible links and take all of them as sure links. The reference set includes 7,037 alignment links. The detailed information about these two sets is described in Table 3.

| Test set | Language | Vocabulary | Words | Average length |
|---|---|---|---|---|
| In-domain | Spanish | 2,998 | 12,369 | 30.9 |
| | English | 2,537 | 11,790 | 29.5 |
| Out-of-domain | Spanish | 2,811 | 10,713 | 27.1 |
| | English | 2,546 | 93,733 | 23.7 |

Table 3. Word Alignment Test Set Statistics

[3]It is located at http://gps-tsc.upc.es/veu/personal/lambert/data/epps-alignref.html.

## Evaluation Metrics

We use the same word alignment evaluation metrics as described in (Och and Ney, 2003). If we use $A$ to indicate the alignments identified by the proposed methods, and $S$ and $P$ to denote the sure and possible links in the reference alignments, the precision, recall, and alignment error rate (AER) are calculated as described in Equations (12), (13) and (14). If we take all links as sure links, then $|P| = |S|$.

$$precision = \frac{|A \cap S|}{|S|} \quad (12)$$

$$recall = \frac{|A \cap P|}{|P|} \quad (13)$$

$$AER = 1 - \frac{|A \cap S| + |A \cap P|}{|A| + |S|} \quad (14)$$

The translation quality was evaluated using a well-established automatic measure: BLEU score (Papineni et al., 2002). And we also use the tool provided in the NAACL/HLT 2006 shared task on SMT to calculate the BLEU scores. We use the same method described in (Koehn and Monz, 2006) to perform the significance test.

## Word Alignment Results

We perform bi-directional (source to target and target to source) word alignments using the GIZA++ toolkit, and obtain the symmetrized alignment results using the six word alignment heuristics described in this paper. The alignment results are shown in Table 4 for the in-domain and out-of-domain test sets.

On both of the test sets, the compromise method "grow-diag" obtains the lowest AER because it makes compromise between precision and recall. The intersection performs the worst because it achieves a much lower recall as compared with other methods. From the results, it can be seen that grow-diag-final, grow-final, and union are recall-oriented methods. Intersection and grow are precision-oriented methods. In general, the compromise method, achieves the best word alignment results, and the precision-oriented method "intersection" gets the worst results.

## Translation Results

We use Koehn's training scripts[4] to train the translation model, and the SRILM toolkit (Stolcke, 2002) to train language model. For translation, we use the Pharaoh decoder (Koehn, 2004). We run the decoder with its default settings.

We use the six word alignment methods described in this paper to get different word alignment results, and then extract phrase pairs consistent with the word alignment results. Table 5 shows the number of the extracted phrase pairs. The intersection method obtains many more phrase pairs, by about a factor of five as compared with the grow-diag-final method. We also compare the detailed information of the phrase pairs. The phrase pairs extracted by grow-diag-final and grow-final includes all phrase pairs extracted by union, and the phrase pairs extracted by

[4] It is located at http://www.statmt.org/wmt06/shared-task/baseline.html.

| Symmetrization strategy | In-domain | | | Out-of-domain | | |
|---|---|---|---|---|---|---|
| | Precision | Recall | AER | Precision | Recall | AER |
| grow-diag-final | 0.7171 | 0.7123 | 0.2851 | 0.6383 | 0.7337 | 0.3173 |
| grow-final | 0.7133 | 0.7142 | 0.2863 | 0.6290 | 0.7346 | 0.3223 |
| union | 0.6915 | **0.7224** | 0.2947 | 0.6143 | **0.7397** | 0.3288 |
| grow-diag | 0.7894 | 0.6817 | **0.2645** | 0.6859 | 0.7052 | **0.3046** |
| grow | 0.8122 | 0.6475 | 0.2701 | 0.7099 | 0.6663 | 0.3126 |
| intersection | **0.8689** | 0.5717 | 0.3064 | **0.7831** | 0.5822 | 0.3321 |

Table 4. Word Alignment Results

| Language pair | grow-diag-final | grow-final | union | grow-diag | grow | intersection |
|---|---|---|---|---|---|---|
| es-en | 37,628,890 | 36,868,632 | 33,249,362 | 99,472,934 | 139,303,869 | 177,741,005 |
| fr-en | 34,511,677 | 33,641,918 | 29,582,174 | 102,494,636 | 142,602,885 | 173,847,297 |
| de-en | 32,954,919 | 31,869,009 | 28,019,951 | 123,086,503 | 166,904,791 | 213,187,373 |

Table 5. The Number of Phrase Pairs

| Language pair | grow-diag-final | grow-final | union | grow-diag | grow | intersection |
|---|---|---|---|---|---|---|
| es-en | 0.3053 | **0.3063** | 0.3042 | 0.3058 | 0.2976 | 0.2892 |
| fr-en | 0.3014 | 0.3020 | 0.3006 | **0.3040** | 0.2964 | 0.2905 |
| de-en | **0.2407** | 0.2397 | 0.2389 | 0.2349 | 0.2288 | 0.2053 |

Table 6. Translation Results on the In-Domain Test Set

| Language pair | grow-diag-final | grow-final | union | grow-diag | grow | intersection |
|---|---|---|---|---|---|---|
| es-en | 0.2479 | 0.2503 | 0.2494 | 0.2516 | **0.2531** | 0.2501 |
| fr-en | 0.1995 | 0.1997 | 0.1979 | **0.2040** | 0.2019 | 0.2022 |
| de-en | 0.1666 | 0.1663 | 0.1643 | **0.1707** | 0.1643 | 0.1530 |

Table 7. Translation Results on the Out-of-Domain Test Set

| Language pair | grow-diag-final | grow-final | union | grow-diag | grow | intersection |
|---|---|---|---|---|---|---|
| es-en | **0.3104** | 0.3100 | 0.3093 | 0.3071 | 0.2988 | 0.2910 |
| fr-en | **0.3084** | 0.3083 | 0.3081 | 0.3047 | 0.2982 | 0.2923 |
| de-en | **0.2428** | 0.2427 | 0.2408 | 0.2360 | 0.2325 | 0.2130 |

Table 8. Translation Results of Filtering on the In-Domain Test Set

| Language pair | grow-diag-final | grow-final | union | grow-diag | grow | intersection |
|---|---|---|---|---|---|---|
| es-en | 0.2608 | 0.2617 | 0.2609 | 0.2634 | **0.2652** | 0.2607 |
| fr-en | 0.2088 | 0.2090 | 0.2112 | **0.2131** | 0.2127 | 0.2128 |
| de-en | 0.1714 | 0.1720 | 0.1692 | **0.1758** | 0.1702 | 0.1598 |

Table 9. Translation Results of Filtering on the Out-of-Domain Test Set

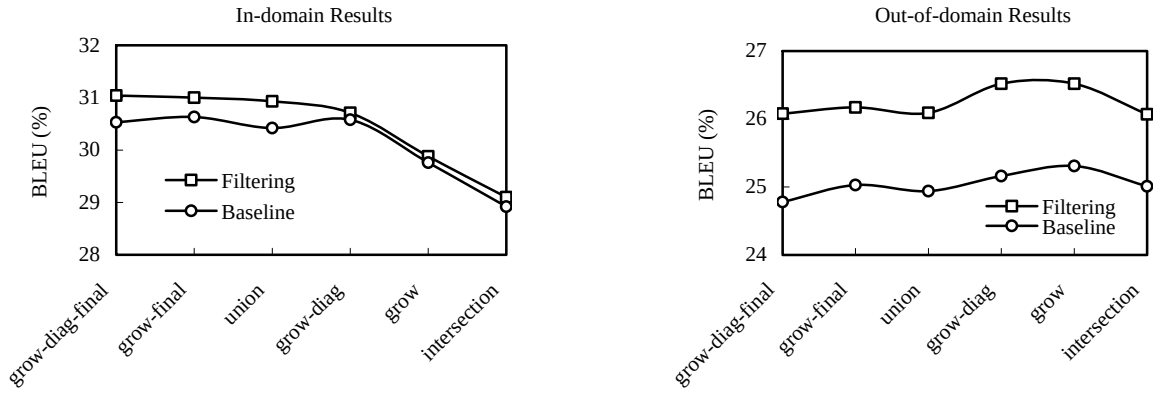


Figure 1. Spanish-English Filtering Results

intersection includes all phrase pairs extracted by grow. More than 90% of phrases pairs extracted by grow-final are covered by grow-diag-final, and more than 95% of phrase pairs extracted by grow-diag are covered by intersection.

The translation results of the different methods on the in-domain test set and out-of-domain test set are shown in Tables 6 and 7, respectively. On both test sets, grow-diag achieves the best BLUE scores for three tasks and the second best for two tasks among six tasks. On the in-domain test set, recall-oriented methods achieve better result than precision-oriented methods. On the out-of-domain test set, the result is very different, where precision-oriented methods achieve better results than recall-oriented methods. [5] This is because precision-oriented methods extract many more phrase pairs, and may cover more source words of the out-of-domain test set. Further analysis shows that, on the out-of-domain test sets, only 355 words (of 29,488) are not covered by the intersection method, while 503 words are not covered by the grow-diag-final method for Spanish to English translation.

### Word Alignment and Translation Quality

As described in the above sections, the grow-diag method that makes compromise between recall and precision performs very well on both in-domain and out-of-domain alignment test sets. Although the six different heuristic methods show a similar tendency for word alignment on both test sets, they perform very differently for translation on the two test sets. Recall-oriented methods perform better on the in-domain test set and precision-oriented methods perform better on the out-of-domain test set. Thus, the relationship between word alignment and machine translation is very complicated. It not only depends on the metrics taken for word alignment and translation quality, and the characteristics of the translation system used (Vilar et al., 2006), it also depends on domains of the corpora investigated on. In conclusion, it is a good idea to use heuristics that make compromise between precision and recall, which can achieve satisfactory translation results on phrase-based SMT on both in-domain and out-of-domain texts.

## Experiments on Noise Filtering

In this section, we perform experiments to filter the phrase pairs used in the Pharaoh system. The training and testing data are the same as those in the translation task.

### Log likelihood Ratio vs. Translation Probability

This section will compare two noise filtering methods: log likelihood ratio and translation probability. Here, we use the grow-diag-final method in French to English translation as a case study. The threshold in equation (7) is set to 0.5 and 0.05 for log likelihood ratio and translation probability, respectively. [6] The filtering results are shown in Table 10. The baseline represents the method before filtering.

From the results, it can be seen that both of the filtering methods outperforms the baseline, with log likelihood ratio performing better. And significance test show that log likelihood ratio significantly outperforms the other methods. Thus, in the following sections, we will only use log likelihood ratio for noise filtering.

| Filtering Method | In-domain | Out-of-domain |
|---|---|---|
| Baseline | 0.3014 | 0.1995 |
| Log likelihood | 0.3084 | 0.2088 |
| Probability | 0.3043 | 0.2030 |

Table 10. Comparison of Filtering Methods

### Filtering Results

Using log likelihood ratio as the filtering method, the translation results after filtering are shown in Tables 8 and 9. In order to directly compare the translation results between the baseline (before filtering) and our filtering method, the results for Spanish to English translation are shown in Figure 1. [7] From the results, it can be seen that the filtering methods is more effective on the out-of-domain test set than on the in-domain test set.

On the in-domain test set, the filtering method is only effective for recall-oriented methods. For the compromise method and precision-oriented methods, the phrase tables are much larger than recall-oriented methods, which may contain much more noise. Log likelihood ratio is not so distinguishable to remove much noise from them.

On the out-of-domain test set, the filtering method is very effective for all of the heuristics, achieving more than 0.01 BLEU score as compared with the baselines. This is because some out-of-domain phrases may occur infrequently in the in-domain training corpus and the phrase translation probability of the infrequently occurring pairs is usually overestimated. Thus, these phrase pairs occurring infrequently may be used for translation. In this case, log likelihood ratio is effective to remove these infrequently occurring pairs, which results in the improvement of translation quality.

### Results by Using Different Sizes of Training Corpus

In order to further analyze the effect of sizes of training corpus, we take Spanish to English translation as a case study. We obtain the training corpora by randomly select 100K, 200k, and 400k sentence pairs from the entire Spanish-English parallel corpus to train translation models. Here we use the three heuristics "grow-diag-final", "intersection", and "grow-diag" to represent the recall-oriented methods, precision-oriented methods, and compromise methods, respectively. The results are shown in Figures 2, 3, and 4. From the figures, it can be seen that, on all sizes of training corpora, the filtering method achieves larger improvement on the out-of-domain test set than on the in-domain test set. On the in-domain test set, filtering is effective for recall-based methods and only achieves minor improvement for other two methods.

[5] In German to English translation, the intersection method achieves lower BLEU score as compared with other methods on both of the test sets. However, the grow method achieves comparable results with other methods on the out-of-domain test set.

[6] The thresholds are set using the development set, which achieves the best results on this set.

[7] The results for the other two translation directions are omitted here because they have similar results as shown in Figure 1.

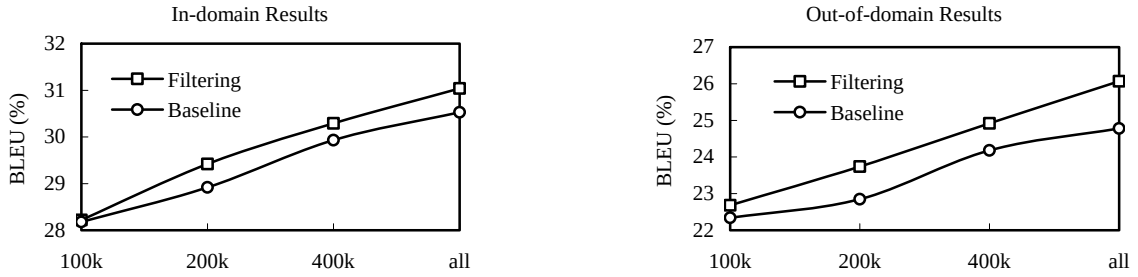


Figure 2. Filtering Results of Grow-diag-final by Using Different Sizes of Training Corpus

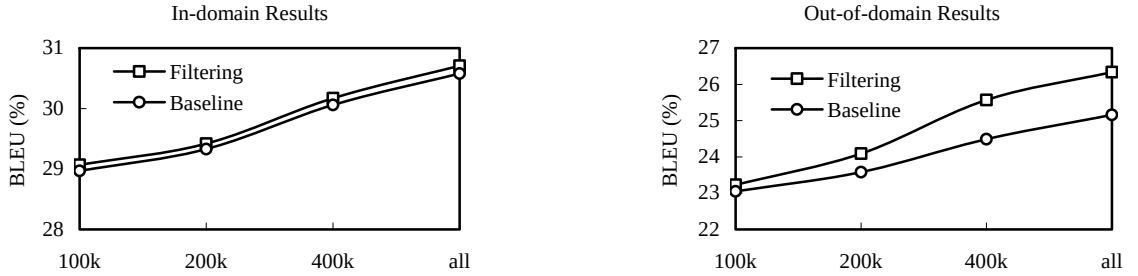


Figure 3. Filtering Results of Grow-diag by Using Different Sizes of Training Corpus

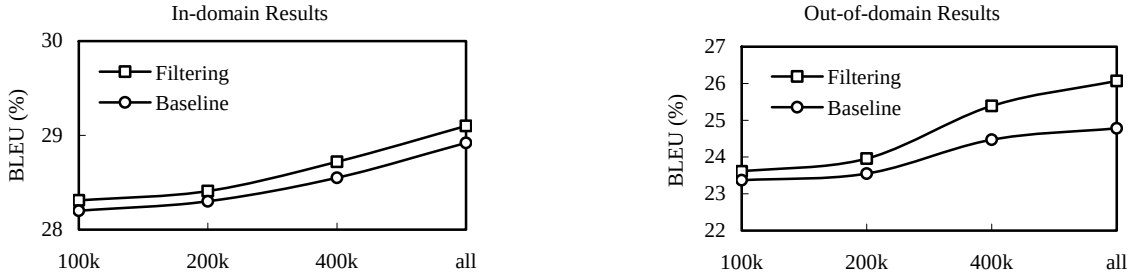


Figure 4. Filtering Results of Intersection by Using Different Sizes of Training Corpus

By using smaller size (100K) of training corpus, filtering the noise is not so effective for both test sets because it is subject to the problem of data sparseness and log likelihood ratio is not so distinguishable to remove them. And by increasing the sizes of the training data, filtering the phrase pairs in the phrase table becomes more effective to achieve gains in translation quality.

## Experiments on Model Combination

In this section, we still use Spanish to English translation as a case study to examine the effect of model combination of the different methods.

### Model Interpolation Vs. Count Merging

Since "grow-diag-final", "grow-diag", and "intersection" represent the three kinds of heuristics, we only perform model combination among these three methods. The combination results are shown in Table 11. All of the methods uses log likelihood ratio to filter the noise in the phrase tables. For the in-domain case, the coefficients are set to 0.8, 0.1, and 0.1 for "grow-diag-final", "grow-diag", and "intersection" for both model interpolation and count merging. For the out-of-domain case, the coefficients are set to 0.1, 0.2, and 0.7 for "grow-diag-final", "grow-diag", and "intersection" for both model interpolation and count merging. All of the weights are tuned on the development set.

The results show that count merging slightly outperforms model interpolation. On the in-domain test set, model combination only slightly improve translation quality while on the out-of-domain test set, model combination significantly improve translation quality. This is because combination of phrase tables extracted using different heuristics does not provide additional information for in-domain translation. In contrast, combining these tables for out-of-domain translation can provide more information to improve translation quality.

| | In-domain | Out-of-domain |
|---|---|---|
| Grow-diag-final | 31.04 | 26.01 |
| Grow-diag | 30.71 | 26.34 |
| Intersection | 29.10 | 26.09 |
| Model interpolation | 31.09 | 26.89 |
| Count merging | 31.12 | 27.01 |

Table 11. Results of Model Combination after Filtering

## Conclusion

This paper evaluated six different word alignment heuristics and their impacts on translation quality. We also investigated the effectiveness of log likelihood ratio to filter the noise in the phrase tables extracted using different heuristic methods and examine the effectiveness of model combination of these methods.

Word alignment results show that using the alignment error rate as a metric, the compromise method performs the best and the precision-oriented method "intersection" performs the worst on both in-domain and out-of-domain test sets.

In the translation task, the results show that the heuristic methods perform differently on the in-domain and out-of-domain test sets. On both of the test sets, the method that makes compromise between precision and recall achieves satisfactory translation quality. On the in-domain test set, the recall-oriented heuristic methods yield better translation quality. On the out-of-domain test set, the precision-oriented heuristic methods yield better translation quality. Thus, the relationship between word alignment and machine translation performance also depends on the domains of training and testing corpora besides other factors such as evaluation metrics and the characteristics of the translation systems.

Results also shows that filtering the noise in the phrase tables results in more improvements of translation quality on the out-of-domain test set than on the in-domain test set. The filtering methods achieve an improvement of about 0.01 BLEU score on the out-of-domain test set.

Model combination results show that count merging performs slightly better than model interpolation on our test sets. And these two methods significantly improve the translation quality on the out-of-domain test set.

## References


Ayan, N.F. and Dorr, B. (2006). Going beyond AER: An Extensive Analysis of Word Alignments and Their Impact on MT. In Proceedings of the Joint Conference of the International Committee on Computational Linguistics and the Association for Computational Linguistics (pp. 9--16).

Bacchiani, M. and Roark, B. (2003). Unsupervised Language Model Adaptation. In Proceedings of International Conference on Acoustic, Speech, and Signal Processing (pp. 224--227).

Bacchiani, M., Roark, B., and Saraclar, M. (2004). Language Model Adaptation with MAP Estimation and the Perceptron Algorithm. In Proceedings of the Human Language Technology Conference and Meeting of the North American Chapter of the Association for Computational Linguistics (pp. 21--24).

Chiang, D. (2005). A Hierarchical Phrase-Based Model for Statistical Machine Translation. In Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (pp. 263--270).

DeNero, J., Gillick, D., Zhang, J., and Klein, D. (2006). Why Generative Phrase Models Underperform Surface Heuristics. In Proceedings of the NAACL 2006 Workshop on Statistical Machine Translation (pp. 3--38).

Dunning, T. (1993). Accurate Methods for the Statistics of Surprise and Coincidence. Computational Linguistics, 19(1), 61--74.

Fraser, A. and Marcu, D. (2006). Measuring Word Alignment Quality for Statistical Machine Translation Technical Report ISI-TR-616, ISI-University of Southern California.

Koehn, P. (2004). Pharaoh: A Beam Search Decoder for Phrase-Based Statistical Machine Translation Models. In Proceedings of the 6th Conference of the Association for Machine Translation in the Americas (pp. 115--124).

Koehn P. (2005). Europarl: A Parallel Corpus for Statistical Machine Translation. In Proceedings of the MT Summit X (pp. 79--86).

Koehn, P. and Monz, C. (2006). Manual and Automatic Evaluation of Machine Translation between European Languages. In Proceedings of the 2006 HLT-NAACL Workshop on Statistical Machine Translation (pp. 102--121).

Koehn, P., Och, F.J., and Marcu, D. (2003). Statistical Phrase-Based Translation. In Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (pp. 127--133).

Lambert, P., Gispert, A., Banchs, R., and Marino, J.B. (2005). Guidelines for Word Alignment Evaluation and Manual Alignment. Language Resources and Evaluation, 39 (4), 267--285.

Lopez A. and Resnik, P. (2006). Word-Based Alignment, Phrase-Based Translation: What's the Link? In Proceedings of the Twelfth Meeting of the Association for Machine Translation in the Americas (pp. 90--99).

Marcu, D. and Wong, W. (2002). A Phrase-Based, Joint Probability Model for Statistical Machine Translation. In Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (pp. 133--139).

Och, F.J. and Ney, H. (2003). A Systematic Comparison of Various Statistical Alignment Models. Computational Linguistics, 29(1), 19--52.

Och, F.J., Tillman, C., and Ney, H. (1999). Improved Alignment Models for Statistical Machine Translation. In Proceedings of the 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (pp. 20--28).

Papineni, K., Roukos, S., Ward, T., and Zhu, W. (2002). BLEU: a Method for Automatic Evaluation of Machine Translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (pp. 311--318).

Stolcke A. (2002). SRILM - An Extensible Language Modelling Toolkit. In Proceeding of the International Conference on Spoken Language Processing (pp. 901--904).

Vilar, D., Popovic, M., and Ney, H. (2006). AER: Do We Need to "Improve" Our Alignments?. In Proceedings of the International Workshop on Spoken Language Processing (pp. 205--212).

Zens, R. and Ney, H. (2004). Improvements in Phrase-Based Statistical Machine Translation. In Proceedings of the Conference on Human Language Technology (pp. 257--264).

Zhao, B. and Waibel, A. (2005). Learning a Log-Linear Model with Bilingual Phrase-Pair Features for Statistical Machine Translation. In Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing (pp. 79--86).